8. (New) The method of claim 3, wherein the first and second data structures are the same data structure.

9. (New) The method of claim 3, wherein the second and third data structures are the same data structure.

10. (New) The method of claim 3, wherein the first and third data structures are the same data structure.

## REMARKS

Claims 1-10 are in the application. Reconsideration of the application is respectfully requested in view of the foregoing amendments and following remarks.

Applicants thank the Examiner for the phone conference of August 30, 2001. During the phone conference, the Slingwine et al. patent (U.S. Patent No. 5,727,209) was discussed.

The Examiner acknowledged that he missed that the Slingwine patent and the present application have a common inventor in Paul McKenney and a common assignee.

### 1) Claims 2-6, as amended, comply with 35 U.S.C. § 112, second paragraph.

Claims 2-6 were rejected as incomplete for omitting essential elements. [*See* Office Action at p. 2.] Applicants respectfully traverse the rejection and request that it be withdrawn.

Applicants have amended claims 2-6 as requested by the Examiner to make it very clear the relation between memory and the data structures.

Consequently, the §112 rejection has been overcome. The Examiner is invited to call the undersigned should any §112 rejections remain.

### 2) Claims 1-6 are allowable under 35 U.S.C. § 103.

Claims 1-6 were rejected under § 103 as unpatentable over the Slingwine reference in view of the Roche reference. [Office Action, p. 4]. Applicant respectfully traverses the rejection and requests that the rejection be withdrawn.

As discussed with the Examiner in the last phone conference, the Slingwine reference has a common inventor to the present application (*i.e.*, Paul McKenney). The Slingwine reference

simply does not teach claim 1, which requires having a "second level bit mask stored in the memory of each processing node" containing a bit indicating whether the processor has not yet passed through the quiescent state (emphasis added). Instead, Slingwine has a single lower level bit mask that is globally accessible. In the present application, by storing the second level bit mask on each processing node, the speed of the overall system is increased. That is, processors on a node can now access the second level bit mask more quickly because it is stored locally.

As discussed during the phone conference, the Roche et al. reference (U.S. Patent No. 4,916,697) does not appear to be even remotely related to the present application. Roche, Col. 8, lines 5-30, which the Examiner cites (Office action, page 4), does not appear to relate to using a bit mask to indicate passing through a quiescent state as argued by the Examiner.

Claim 2 requires " a variable stored in the physical memory of each node and containing the current generation number." (Emphasis added). Thus, the current generation number is stored on each node allowing faster access by the processors.

Claim 3 requires "determining from a data structure on the processor's node ...." Again, the data structure is stored on the processor's node.

Claims 4-10 are dependent claims and should also be in condition for allowance for the reasons stated above.

## CONCLUSION

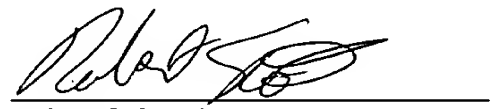
Claims 1-10 are in condition for allowance. Such action is respectfully requested.

## REQUEST FOR INTERVIEW

If the Examiner finds that the amendment does not make the application allowable, the Examiner is formally requested to contact the undersigned attorney at (503) 226-7391 prior to issuance of the next communication in order to arrange a telephonic interview. It is believed that a brief discussion of the merits of the present application will allow the application to be passed to issue. Applicant submits the foregoing remarks so that the Examiner may fully evaluate Applicant's position, thereby enabling the interview to be more productive.

Respectfully submitted,

KLARQUIST SPARKMAN, LLP

By 

Robert F. Scotti
Registration No. 39,830

One World Trade Center, Suite 1600
121 S.W. Salmon Street
Portland, Oregon 97204
Telephone: (503) 226-7391
Facsimile: (503) 228-9446

**Marked-up Version of Amended Claims**
**Pursuant to 37 C.F.R. §§ 1.121(b)-(c)**

In the Claims:

Please amend claims 2 and 3 as follows:

2. (Amended) In a multiprocessor computer system having multiple interconnected processing nodes each with one or more processors and physical memory, a data structure stored in the physical memory on each node for storing a number of the current generation of data elements being processed by a processor on a node, the data structure comprising a variable stored in the physical memory of each node and containing the current generation number, wherein the current generation number on the nodes are updated in lockstep so that the nodes have local access to copies of the current generation number.

3. (Amended) In a multiprocessor computer system having multiple interconnected processing nodes each with one or more processors and physical memory, a method for a processor to maintain a summary of thread activity as part of a method for providing mutual exclusion between current and next generation data elements, comprising:

determining from a first data structure on the processor's node if the processor has passed through a quiescent state;

if so, determining from a second data structure on the processor's node if all other processors on its node have passed through a quiescent state; and

if so, indicating in a third data structure accessible to all nodes that all processors on the processor's node have passed through a quiescent state.

4. (Amended) The method of claim 3 wherein if the processor determines from the first data structure on the processor's node that the processor has not passed through a quiescent state, having a callback processor check if the processor has passed through a quiescent state and, if so, having the processor indicate in the first data structure that it has passed through a quiescent state.

5. (Amended) The method of claim 3 wherein if the processor determines from the third data structure accessible to all nodes that the processor is the last processor to pass through

a quiescent state, having the processor update a fourth data structure stored in the memory of each node for storing a number of the current generation of elements being processed on the node.

6. (Amended) The method of claim 3 wherein if the processor determines from the third data structure accessible to all nodes that it is the last processor to pass through a quiescent state, having a callback processor determine if there are callbacks waiting for a subsequent generation, and, if so, updating the first data structure on each node and the third data structure accessible to all nodes to indicate that all processors need to pass through an additional quiescent state.

7. (New) The method of claim 3, wherein the first, second, and third data structures are the same data structure.

8. (New) The method of claim 3, wherein the first and second data structures are the same data structure.

9. (New) The method of claim 3, wherein the second and third data structures are the same data structure.

10. (New) The method of claim 3, wherein the first and third data structures are the same data structure.